



1. Giới thiệu khóa học lập trình hướng đối tượng với Python

2. Cách cài đặt Python và Pycharm IDE

3. Chương trình Python - Hello World

4. Hàm main trong Python

5. Biến trong Python

6. Chuỗi trong Python

7. Python TUPLE

8. Từ điển trong Python (python dict)

9. Toán tử trong Python

10. Hàm trong Python



11. Python if, else

12. Vòng lặp For và While trong Python

13. Python OOP

14. Biểu thức chính quy trong Python - Python regex

15. Python DateTime

16. Hướng dẫn sử dụng Python CALENDAR

< Bài học trước

Bài kế tiếp >



Vimentor Admin



THÊM VÀO KÊNH

THEO DÕI

## Tệp CSV là gì?

Tệp CSV là một loại tệp văn bản thuần túy sử dụng cấu trúc cụ thể để sắp xếp dữ liệu dạng bảng. CSV là một định dạng phổ biến để trao đổi dữ liệu vì sự nhỏ gọn, đơn giản và tính khái quát. Nhiều dịch vụ trực tuyến cho phép người dùng xuất dữ liệu dạng bảng từ trang web thành tệp CSV. Các tệp của CSV sẽ mở bằng Excel và gần như tất cả các cơ sở dữ liệu đều có công cụ cho phép nhập từ tệp CSV. Định dạng chuẩn được xác định bởi dữ liệu hàng và cột. Hơn nữa, mỗi hàng được kết thúc bởi một dòng mới để bắt đầu hàng tiếp theo. Cũng trong hàng, mỗi cột được phân tách bằng dấu phẩy.

Trong bài này, bạn sẽ tìm hiểu về:

- Tệp CSV là gì?
- Tệp CSV mẫu
- Mô-đun CSV trong Python
- Các hàm trong mô-đun CSV
- Đọc từ tệp CSV
- Đọc dưới dạng một từ điển
- Ghi vào tệp CSV
- Đọc tệp CSV với Pandas
- Ghi vào tệp CSV bằng Pandas



## Tệp CSV mẫu

Dữ liệu ở dạng bảng cũng được gọi là CSV (Comma separated values)- nghĩa là "giá trị được phân tách bằng dấu phẩy". Đây là một định dạng văn bản dành cho việc trình bày dữ liệu dạng bảng. Mỗi dòng của tệp là một dòng của bảng. Các giá trị của các cột riêng lẻ được phân tách bằng dấu phân cách - dấu phẩy (,), dấu chấm phẩy (;) hoặc ký hiệu khác. CSV có thể dễ dàng đọc và xử lý bởi Python.

## Bảng dữ liệu

Ngôn ngữ lập trình		Thiết kế bởi	Xuất hiện	Phần mở rộng
Python		Guido van Rossum	1991	.py
Java		James Gosling	1995	.java
C ++		Bjarne Stroustrup	1983	.cpp

Bạn có thể biểu diễn bảng này trong csv như dưới đây.

## Dữ liệu CSV

Ngôn ngữ lập trình,Thiết kế bởi, Xuất hiện,Định dạng mở rộng

Python, Guido van Rossum, 1991, .py

Java, James Gosling, 1995, .java

C ++, Bjarne Stroustrup, 1983, .cpp



Như bạn có thể thấy mỗi hàng là một dòng mới và mỗi cột được phân tách bằng dấu phẩy. Đây là một ví dụ cho thấy cách tệp CSV được bố trí.

## Mô-đun CSV trong Python

Python cung cấp một mô-đun CSV để xử lý các tệp CSV. Để đọc / ghi dữ liệu, bạn cần duyệt qua các hàng của CSV. Bạn cần sử dụng phương pháp tách để lấy dữ liệu từ các cột được chỉ định.


### Các hàm trong mô-đun CSV

Trong tài liệu mô-đun CSV, bạn có thể tìm thấy các hàm sau:

- `csv.get_dialect('d_size_limit')` - trả lại kích thước trường tối đa
- `csv.get_dialect()` - lấy dữ liệu được liên kết với tên



- `csv.reader` - đọc dữ liệu từ tệp csv
- `csv.register_dialect` - liên kết dữ liệu với tên
- `csv.writer` - ghi dữ liệu vào tệp csv
- `csv.unregister_dialect` - xóa dữ liệu liên quan đến tên đăng ký
- `csv.QUOTE_ALL` - Trích dẫn mọi thứ, không phân biệt kiểu.
- `csv.QUOTE_MINIMAL` – Trích dẫn các trường chứa ký tự đặc biệt
- `csv.QUOTE_NONNUMERIC` - Trích dẫn tất cả các trường không có giá trị số
- `csv.QUOTE_NONE` - Không trích dẫn bất cứ điều gì ở đầu ra

Trong bài này, chúng ta sẽ chỉ tập trung vào các chức năng đọc và ghi cho phép bạn chỉnh thay đổi và thao tác dữ liệu trong tệp CSV. 

## Cách đọc tệp CSV

Để đọc dữ liệu từ tệp CSV, bạn phải sử dụng hàm đọc để tạo đối tượng đọc.

Hàm đọc được viết để lấy từng hàng của tệp và tạo danh sách tất cả các cột. Sau đó, bạn phải chọn cột bạn muốn trích xuất dữ liệu.

Điều này nghe có vẻ phức tạp hơn nhiều so với thực tế vốn có của nó. Hãy xem ví dụ sau để thấy rằng làm việc với tệp csv không quá khó.

```
#import necessary modules
import csv
with open('X:\data.csv', 'rt') as f:
    data = csv.reader(f)
    for row in data:
        print(row)
```

Khi bạn thực hiện chương trình trên, đầu ra sẽ là:

```
['Pr', 'ing language; Designed by; Appeared; Extension']
['Pyt', 'Guido van Rossum; 1991; .py']
['Java; James Gosling; 1995; .java']
```

## Cách đọc CSV dưới dạng từ điển

Bạn cũng có thể sử dụng DictReader để đọc tệp CSV. Các kết quả được xem như một từ điển trong đó hàng tiêu đề là khóa và các hàng khác là giá trị.

Hãy xem xét các mã sau đây

```
#import necessary modules
import csv

reader = csv.DictReader(open("file2.csv"))
for raw in reader:
    print(raw)
```

Kết quả của mã này là:

```
OrderedDict([('Programming language', 'Python'), ('Designed by', 'Guido van Rossum'), (' Appeared', ' 1991'), (' Extension', ' .py')])
OrderedDict([('Programming language', 'Java'), ('Designed by', 'James Gosling'), (' Appeared', ' 1995'), (' Extension', ' .java')])
OrderedDict([('Programming language', 'C++'), ('Designed by', ' Bjarne Stroustrup'), (' Appeared', ' 1985'), (' Extension', ' .cpp')])
```

```
1 #import necessary modules
2 import csv
3 reader = csv.DictReader(open("X:\data.csv"))
4 for raw in reader:
5     print(raw)
6
```

Run: C:\Users\Admin\PycharmProjects\CSV\venv\Scripts\python.exe C:/Users/Admin/.PyCharmCE2018.2/config/scratches/scratch.py

```
OrderedDict([('Programming language', 'Python'), ('Designed by', 'Guido van Rossum'), ('Appeared', '1991'), ('Extension', '.py')])
OrderedDict([('Programming language', 'Java'), ('Designed by', 'James Gosling'), ('Appeared', '1995'), ('Extension', '.java')])
OrderedDict([('Programming language', 'C++'), ('Designed by', 'Bjarne Stroustrup'), ('Appeared', '1983'), ('Extension', '.cpp')])
```

Process finished with exit code 0

Sử dụng cách này để đọc dữ liệu từ tệp CSV dễ dàng hơn nhiều so với phương pháp trước đó. Tuy nhiên, đây không phải là cách tốt nhất để đọc dữ liệu.

## Cách viết tệp CSV

Khi bạn có một bộ dữ liệu mà bạn muốn lưu trữ trong tệp CSV, bạn phải sử dụng hàm `writer()`. Để lặp lại dữ liệu qua các hàng (dòng), bạn phải sử dụng hàm `writerow()`.

dấu nhảy đơn.

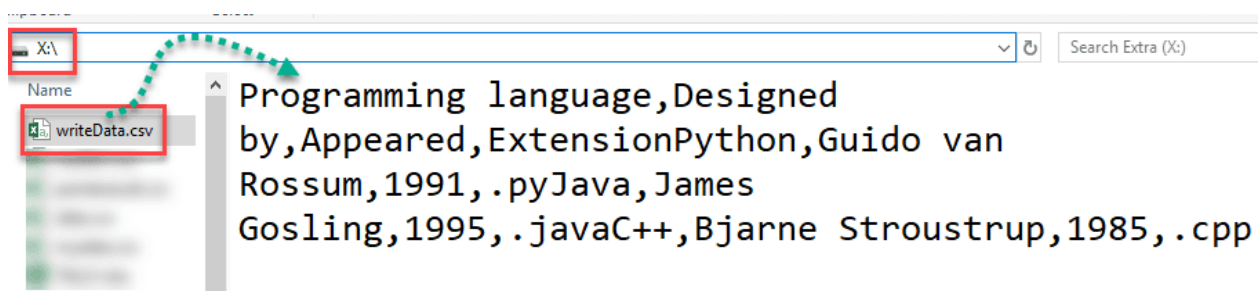
```
#import necessary modules
import csv

with open('X:\writeData.csv', mode='w') as file:
    writer = csv.writer(file, delimiter=',', quotechar='"',
quoting=csv.QUOTE_MINIMAL)

    #way to write to csv file
    writer.writerow(['Programming language', 'Designed by', 'Appeared',
'Extension'])
    writer.writerow(['Python', 'Guido van Rossum', '1991', '.py'])
    writer.writerow(['Java', 'James Gosling', '1995', '.java'])
    writer.writerow(['C++', 'Bjarne Stroustrup', '1985', '.cpp'])
```

Kết quả trong tệp csv là:

```
Programming language, Designed by, Appeared, Extension
Python, Guido van Rossum, 1991, .py
Java, James Gosling, 1995, .java
C++, Bjarne Stroustrup, 1985, .cpp
```



## Đọc tệp CSV với Pandas

Pandas là một thư viện mã nguồn mở cho phép bạn xử lý dữ liệu trong Python. Pandas cung cấp một cách dễ dàng để tạo, thao tác và xóa dữ liệu.

Bạn phải cài đặt thư viện pandas bằng lệnh `pip install pandas`. Trong windows, bạn sử dụng `cmd` thì lệnh này trong cửa sổ dòng lệnh còn trong Linux bạn cần chạy nó bằng Terminal.

Đọc CSV vào một Pandas DataFrame rất nhanh chóng và dễ dàng.

```
import pandas
result = pandas.read_csv('X:\data.csv')
print(result)
```

Kết quả:

```
Programming language, Designed by, Appeared, Extension
0 Python, Guido van Rossum, 1991, .py
1 Java, James Gosling, 1995, .java
2 C++, Bjarne Stroustrup, 1983, .cpp
```

Thư viện này rất hữu ích. Chỉ với ba dòng mã bạn có kết quả giống như trước đó. Pandas biết rằng dòng đầu tiên của CSV chứa tên cột và nó sẽ tự động sử dụng chúng.

## Ghi vào tệp CSV bằng Pandas

Ghi vào tệp CSV bằng Pandas dễ như đọc. Trước tiên, bạn phải tạo DataFrame dựa trên mã sau đây.



```
from pandas import DataFrame
C = {'Programming language': ['Python', 'Java', 'C++'],
     'Designed by': ['Guido van Rossum', 'James Gosling', 'Bjarne Stroustrup'],
     'Appeared': ['1991', '1995', '1985'],
     'Extension': ['.py', '.java', '.cpp'],
     }
df = DataFrame(C, columns= ['Programming language', 'Designed by', 'Appeared',
                             'Extension'])
export_csv = df.to_csv (r'X:\pandaresult.csv', index = None, header=True) # here you
have to write path, where result file will be stored
print (df)

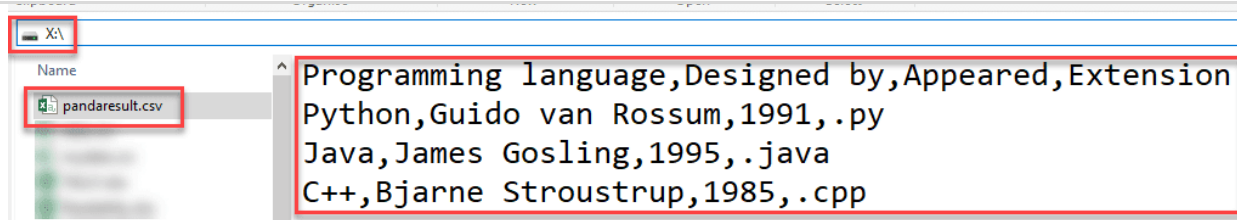
in (df)
```

Đây là đầu ra

```
Programming language, Designed by, Appeared, Extension
0 Python, Guido van Rossum, 1991, .py
1 Java, James Gosling, 1995, .java
2 C++, Bjarne Stroustrup, 1983, .cpp
```







## Tổng kết

Giờ bạn đã biết cách sử dụng hàm 'csv' cũng như việc đọc và ghi dữ liệu ở định dạng CSV. Các tệp CSV được sử dụng rộng rãi trong các ứng dụng phần mềm vì chúng dễ đọc, dễ quản lý. Thêm vào đó, kích thước nhỏ của chúng khiến việc xử lý và truyền đi tương đối nhanh.

Mô-đun csv cung cấp các chức năng và các lớp khác nhau cho phép bạn đọc và ghi dễ dàng. Bạn có thể xem tài liệu Python chính thức để tìm thấy một số mẹo và mô-đun thú vị hơn. CSV là cách tốt nhất để lưu, xem và gửi dữ liệu. Sự thật là việc học nó không khó như bạn nghĩ ban đầu. Và chỉ với việc luyện tập một chút, bạn sẽ dễ dàng thành thạo nó.



Pandas là một thay thế tuyệt vời để đọc các tệp CSV.

Ngoài ra, có nhiều cách khác để phân tích các tệp văn bản với các thư viện như ANTLR, PLY và PlyPlus. Tất cả chúng đều có thể xử lý phân tích cú pháp phức tạp. Trong trường hợp các thao tác với chuỗi đơn giản không thành công, bạn có thể sử dụng các biểu thức chính quy.

[In](#)[PDF](#)[< Bài học trước](#)[Bài kế tiếp >](#)

### KHÓA HỌC LIÊN QUAN

[Lập trình C](#)[Lập trình C++](#)[Khóa học kỹ thuật thiết kế và triển khai hệ thống với Python](#)[Machine Learning và ứng dụng](#)

 Không thích: 0

 Bình luận: 0

Viết bình luận...

\*\* Nếu bạn muốn viết các nội dung đặt biệt thì hãy làm theo [hướng dẫn sau](#)

## VỀ VIMENTOR

[Điều khoản sử dụng](#)

[Chính sách bảo mật](#)

[Tuyển dụng](#)



## HỖ TRỢ

[Viết blog](#)

[Viết đánh giá phỏng vấn](#)

[Trở thành tác giả của vimentor](#)

## LIÊN HỆ VỚI CHÚNG TÔI

 Tân Mỹ - Mỹ Đình 1 - Nam Từ Liêm - Hà Nội



 [admin@vimentor.com](mailto:admin@vimentor.com)

Bản quyền thuộc về Vimentor © 2017. Tất cả các quyền được bảo hộ

